TECHNICAL RESEARCH REPORT

A Local Pursuit Strategy for Bio-Inspired Optimal Control with Partially-Constrained Final State

by Cheng Shao, Dimitrios Hristu-Varsakelis

TR 2005-76

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **2005**	2. REPORT TYPE	3. DATES COVERED **-**

4. TITLE AND SUBTITLE **A Local Pursuit Strategy for Bio-Inspired Optimal Control with Partially-Constrained Final State**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Army Research Office,PO Box 12211,Research Triangle Park,NC,27709**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**12**	

A Local Pursuit Strategy for Bio-Inspired Optimal Control with Partially-Constrained Final State [⋆]

Cheng Shao [a], Dimitrios Hristu-Varsakelis [a,*]

[a] *Department of Mechanical Engineering and Institute for Systems Research*
University of Maryland, College Park, MD 20742 USA

Abstract

Inspired by the process by which ants gradually optimize their foraging trails, this report investigates the cooperative solution of a class of free-final time, partially-constrained final state optimal control problems by a group of dynamic systems. A class of cooperative, pursuit-based algorithms are proposed for finding optimal solutions by iteratively optimizing an initial feasible control. The proposed algorithms require only short-range, limited interactions between group members, avoid the need for a "global map" of the environment on which the group evolves, and solve an optimal control problem in "small" pieces, in a manner which will be made precise. The performance of the algorithms is illustrated in a series of simulations and laboratory experiments.

Key words: Co-operative control, Optimization, Algorithms, Agents, Group work, Trajectories, Minimum-time control

1 Introduction

In recent years, problems in cooperative control are increasingly capturing the attention of researchers, fueled by the development of decentralized control systems with cost and performance advantages. The rising interest in deploying cooperative systems also stems from their potential to perform tasks that are not feasible for individuals. Examples include remote exploration and information gathering by swarms of small autonomous robots [1], and satellite arrays, to name a few. Members of such "engineered collectives" usually have – just like their natural counterparts – limited sensing, communication and computing capabilities. This suggests that each member can only perform relatively simple tasks. However, individual limitations can often be overcome by cooperation, if one can identify an effective way to organize the group into "more than the sum of its parts". Doing so may be difficult because it requires decomposing a desired group behavior into individual behaviors. The results however, can be spectacular, as is often demonstrated by biological collectives. For example, a school of fish can coordinate their movement in a tight formation and respond almost as fast as a single organism to evade encountering dangers; worker honey bees share information by "dancing" and distribute themselves among nectar sources in accordance with the profitability of each source; ants are known to utilize pheromone secretions for recruiting nest-mates and for optimizing their foraging trails [4]. Observations of such activities in nature have already seeded a variety of research, from modeling of animal group behaviors [4,2,15,9], to distributed collective covering and searching [16,12], cooperative estimation [13,10], cooperative robotic teams [6,17,11] and biologically-motivated optimization [5,3].

A particularly interesting example of cooperation in natural animal aggregates has to do with the foraging activity of ant colonies. Ants recruit their co-workers to convey food back to the nest when they find it. Finding an efficient (short) path between the nest and food source appears to be too complicated for individual ants to accomplish, considering their limited cognition and size relatively to the obstacles in the environment, including stones, sticks and crevices. Nonetheless, a colony of ants exhibit a high degree of competence in such tasks [4].

[⋆] This work was supported by the National Science Foundation under Grant No. EIA0088081 and by ARO ODDR&E MURI01 Grant No. DAAD19-01-1-0465, (Center for Communicating Networked Control Systems, through Boston University).

[*] Corresponding author. Tel: +1-301-405-5283, Fax: +1-301-314-9477.

Email addresses: cshao@glue.umd.edu (Cheng Shao), hristu@umd.edu (Dimitrios Hristu-Varsakelis).

Several models have been proposed in the attempt to capture the organizing principle by which ants find shortest paths when foraging. For example, [4] described a model based on the use of pheromonal secretions that help ants choose trails. Briefly, pheromonal secretions are laid along the paths by ants to recruit nestmates and to indicate the frequency of use for that path. Inspired by that model, [16] developed robust adaptive algorithms to perform tasks requiring the traversal of an unknown region, such as cleaning the floor of an unmapped building; [5] introduced a search methodology based on the "distributed autocatalytic process" to solve a classical optimization problem, the traveling salesman problem.

A particularly simple – but elegant – ant colony organizing rule was presented in [2], where it was shown that ants that "pursued" one another on $\mathbb{R}^2$ (each pointing its velocity vector towards a predecessor) had the effect of producing progressively "straighter" trails. That idea was later extended to path optimization problems involving kinematic vehicles in non-Euclidean environments [7,8].

Although local pursuit was inspired from observations of ant colonies and applied to other engineered collectives, the last works in [2,7] dealt exclusively with the "discovery" of geodesics, meaning that the autonomous system-members of the group had simple dynamics ($\dot{x} = u$) with no drift terms. In [8], it was shown that the earlier work could be generalized to a much broader class of optimal control problems, and collectives whose members have non-trivial dynamics. The proposed algorithm, termed "local pursuit" (to use the term coined in [2]), guides members of a group toward the solution of an optimal control problem. However, the algorithms presented in [8] were restricted to problems with fixed final time and fixed final states. This report explores a modified version of local pursuit for solving a broader and more interesting class of optimal control problems with free final time and partially-constrained final states.

Under our proposed control strategy, members of the collective do not need a global map of their environment or even an agreed-upon common coordinate system. Thus, powerful sensing and mass information exchanging are not needed, neither is the computation over "long" distances. This makes the proposed algorithms most useful in trajectory optimization problems which are easier to solve when boundary conditions are "close" to one another (because of, for example, the members' computational or sensing limitations), with the term "close" taken to include not only geographical separation but also distance on the manifold on which copies of a dynamical system evolve.

The remainder of this report is organized as follows: Section 2 describes the optimal control problems to be addressed and proposes an iterative algorithm that is appropriate for a group of cooperating dynamical systems. Section 3 discusses the main results concerning the performance of the proposed algorithm. Section 4 presents a series of simulations and laboratory experiments that illustrate our approach.

2 A bio-inspired algorithm for optimal control

We are interested in the solution of optimal control problems using a group of cooperating "agents". The term "agent" will refer to a member of a group of dynamical systems, each taken to be a copy of:

$$\dot{x}_k = f(x_k, u_k),\ x_k(t) \in \mathbb{R}^n, u_k(t) \in \Omega \subset \mathbb{R}^m \qquad (1)$$

for $k = 0, 1, 2 \ldots$. Physically, each copy of (1) could stand for a robot, UAV or other autonomous system.

2.1 Problem Statement and Notation

The problem under consideration is:

Problem 1 *Find a trajectory $x^*(t)$, a final time $\Gamma^* > 0$ and a final state $x^*(\Gamma^*)$ that minimize*

$$J(x, \dot{x}, t_0) = \int_{t_0}^{t_0+\Gamma} g(x, \dot{x})dt + F(x(t_0 + \Gamma)) \qquad (2)$$

subject to the constraints $x(t_0) = x_0$ and $Q(x(t_0 + \Gamma)) = 0$,

where it is assumed that $g(x(t), \dot{x}(t)) \geq 0$, $F(x(t_0 + \Gamma)) \geq 0$ and that $Q(\cdot)$ is an algebraic function of the state.

Definition 1 *Given the final state constraint $Q(x) = 0$, the* constraint set *of x is*

$$S_Q \triangleq \{x | Q(x) = 0\}.$$

The function $F(x)$ in (2) will be taken to be of the form:

$$G(x) = \begin{cases} F(x) & \text{if } x \in S_Q \\ 0 & \text{if } x \notin S_Q \end{cases}$$

with $F(x) \geq 0, \forall x \in S_Q$. Problem 1 involves optimal control with free final time and partially-constrained final state. Fixed final state problems, where S_Q is a single state [14,8], are special cases of what are considered here.

For any pair of fixed states $a, b \in \mathbb{D} \subset \mathbb{R}^n$, let $x^*(t)$ denote the optimal trajectory from a to b with free final time (minimizing J with respect to x and Γ only). The corresponding optimal final time is $\Gamma^*(a,b)$. The cost of following x^* is denoted as:

$$\begin{aligned}\eta(a,b,t_0) &\triangleq \int_{t_0}^{t_0+\Gamma^*} g(x^*,\dot{x^*})dt + G(x^*(t_0+\Gamma^*)) \\ &= \min_{x,\Gamma} J(x,\dot{x},t_0) \qquad (3)\end{aligned}$$

subject to $x(t_0)=a,\ x(t_0+\Gamma)=b$.

Now, let $x^*(t)$ be the optimal trajectory from an initial state a to the constraint set S_Q, and let $\Gamma_Q^*(a,S_Q)$ be the corresponding optimal final time from a to S_Q. The cost of following x^* is denoted by

$$\begin{aligned}\eta_Q(a,t_0) &\triangleq \int_{t_0}^{t_0+\Gamma_Q^*} g(x^*,\dot{x^*})dt + G(x^*(t_0+\Gamma_Q^*)) \\ &= \min_{x,\Gamma_Q} J(x,\dot{x},t_0) \qquad (4)\end{aligned}$$

subject to $x(t_0)=a, Q(x(t_0+\Gamma_Q))=0$.

The cost of following a generic trajectory $x(t)$ of (1) during $[t_0,t_0+\sigma)$ is denoted by:

$$C(x,t_0,\sigma) \triangleq \int_{t_0}^{t_0+\sigma} g(x,\dot{x})dt + G(x(t_0+\sigma)) \qquad (5)$$

The following facts can be derived easily from the properties of optimal trajectories and will be helpful in the sequel:

Fact 2 *Let η,η_Q,C as defined in (3),(4),(5), and let $x_k(t)$ be a generic trajectory of (1). Then, the following hold:*

(1) $\eta(a,b,t_0) \le C(x_k,t_0,\Gamma)$ for any $x_k(\cdot)$ with $x_k(t_0)=a,\ x_k(t_0+\Gamma)=b$.
(2) $\eta(a,c,t_0) \le \eta(a,b,t_0)+\eta(b,c,t_0+\sigma)$ with $\sigma=\Gamma^*(a,b)$.
(3) $\eta_Q(a,t_0) \le \eta(a,b,t_0)$ for any $b \in S_Q$.

2.2 Algorithm

Assume that there is available an initial feasible (but suboptimal) control/trajectory pair $(u_{feas}(t),x_{feas}(t))$ for (1), obtained through a combination of a-priori knowledge about the problem and/or random exploration. Following the idea in [2,8], the agents are scheduled to leave the initial state x_0 sequentially and pursue one another towards the set S_Q, in a way which will be made precise shortly. The sequence is initiated with the first agent following x_{feas} to reach a point in S_Q. Each subsequent agent will attempt to intercept its predecessor – along optimal trajectories defined by (3) – if the predecessor has not reached its final state in S_Q. If the predecessor has already reached the constraint set S_Q, then the pursuer ignores the preceding agent and instead evolves along the optimal trajectory defined by (4). The precise rules that govern the movement of each agent are:

Algorithm 1 *(Modified Continuous Local Pursuit): Identify the starting state x_0 on $\mathbb{D}$ and the constraint set S_Q. Let $x_0(t)$ ($t\in[0,T_0]$) be an initial trajectory satisfying (1) with $x_0(0)=x_0,\ Q(x_0(T_0))=0$. Choose $0<\Delta\le T_0$.*

(1) For $k=1,2,3\ldots$, let $t_k=k\Delta$ be the starting time of k^{th} agent. Let $u_k(t)=0, x_k(t)=x_0$ for $0\le t\le t_k$.
(2) For all $t\ge t_k$, calculate $u_t^*(\tau)$ for all $t\in[t_k,t_k+T_k]$ such that $f(\hat{x}_k(\tau),u_t^*(\tau))=\dot{\hat{x}}_t(\tau)$, and $\hat{x}_t(\tau)$ achieves
$$\begin{cases}\eta(x_k(t),x_{k-1}(t),t), & \text{if } x_{k-1}(t)\notin S_Q \\ \eta_Q(x_k(t),t), & \text{if } x_{k-1}(t)\in S_Q\end{cases}$$
where $\tau\in[t,t+\Gamma^*(x_k(t),x_{k-1}(t))]$ if $x_{k-1}(t)\notin S_Q$ or $\tau\in[t,t+\Gamma_Q^*(x_k(t),S_Q)]$ if $x_{k-1}(t)\in S_Q$
(3) Apply $u_k(t)=u_t^*(0)$ to the k^{th} agent.
(4) Repeat from step 2, until the k^{th} agent reaches S_Q.

When discussing pairs of agents during pursuit, the $(k-1)^{th}$ agent is designated as the "leader" and the k^{th} agent as the "follower". As Step 2 of the algorithm indicates, there are two types of follower movements, "catching up" and "free running", depending on whether the leader has reached the final constraint set S_Q. The former type lets agents "learn" from their leaders, while the "free running" stage enables them to find the optimal final state within S_Q once they are close enough to that set. Both stages will be essential in order for the group to solve Problem 1.

Note that modified continuous local pursuit (mCLP) requires each follower to continuously update its movement (via sensing and computing) to catch up with its leader during the pursuit process. Continuous pursuit may imply a significant computational burden for each agent, especially in cases where the optimal trajectories "linking" follower and leader cannot be written down in closed form. For instances of Problem 1 where for each follower the optimal time to reach the leader is lower bounded for all time, then it is possible to alter the previous algorithm so that each agent only performs a finite number of updating as it evolves from x_0 to S_Q. This is done by defining a modified "sampled local pursuit" policy, similar to that used in [8] for fixed final state problems:

Algorithm 2 *(Sampled Local Pursuit): Identify the starting state x_0 on $\mathbb{D}$ and the constraint set S_Q. Let $x_0(t), t\in[0,T_0]$ be an initial trajectory satisfying (1) with $x_0(0)=x_0,\ Q(x_0(T_0))=0$. Choose the pursuit interval Δ such that $0<\Delta\le T_0$.*

(1) For $k = 1, 2, 3 \ldots$, let $t_k = k\Delta$ be the starting time of the k^{th} agent, i.e. $u_k(t) = 0,\ x_k(t) = x_0$ for $0 \leq t \leq t_k$.

(2) Choose the updating interval $\delta_i < \min(\Delta, \Gamma^*_{i-1})$, where Γ^*_{i-1} is the optimal final time of the last update defined by Eq. (3) or (4), and denote $\Gamma^*_{-1} = \Delta$ for convenience. When $t^i_k = t^{i-1}_k + \delta_{i+1}, t^0_k = t_k, i = 0, 1, 2, 3, \ldots$, calculate the control $u^*_t(\tau)$ that achieves (subj. to (1)):

$$\begin{cases} \eta(x_k(t), x_{k-1}(t), t), & \text{if } x_{k-1}(t) \notin S_Q \\ \eta_Q(x_k(t), t), & \text{if } x_{k-1}(t) \in S_Q \end{cases}$$

where $\tau \in [t, t + \Gamma^*(x_k(t), x_{k-1}(t))]$ if $x_{k-1}(t) \notin S_Q$ or $\tau \in [t, t + \Gamma^*_Q(x_k(t), S_Q)]$ if $x_{k-1}(t) \in S_Q$

(3) Apply $u_k(t) = u^*_{t^i_k}(t - t^i_k)$ to the k^{th} agent for $t \in [t^i_k, t^{i+1}_k)$.

(4) Repeat from step 2 until the k^{th} agent reaches S_Q.

Under modified sampled local pursuit (mSLP) each agent executes a finite number of "updates" of its trajectory, once every $\delta < \Delta$ time units. mSLP's reduced computational demands make it attractive in cases where the complexity of the agents' dynamics as well as that of the environment they evolve in make necessitate the use of numerical methods for finding optimal trajectories. In fact, the sampled version of local pursuit algorithm can itself be useful as a numerical method for computing optimal controls. A full analysis of local pursuit in that light is currently under way.

In the algorithms defined above, we assume that each follower does not intercept its leader. If an interception does occur, the follower will "join" its leader by repeating the leader's trajectory after the time of interception. Because the initial agent travels along its trajectory for T_0 units of time and the pursuit interval Δ is finite, there will be a finite number of such events whose existence will not affect the results discussed below.

3 Main Results

In this section we explore the behavior of the group (1) under continuous local pursuit (mCLP). Although we will not do so here, similar results can be derived for sampled local pursuit (mCLP), using [8] as a starting point.

mCLP defines an ordered sequence of trajectories $\{x_k(t)\}$. This section will first investigate the convergence of the trajectories' cost, and then will show that the trajectories themselves converge to a local optimum. It will be convenient to distinguish between the *planned trajectory*, denoted by $\hat{x}(t)$, that a follower computes at every point in time in order to reach its leader, and the *realized trajectory*, denoted by $x(t)$, along which the follower actually evolves.

Lemma 1 *Consider a leader-follower pair evolving under mCLP with a pursuit interval Δ. Let the leader's trajectory be $x_{k-1}(t)$ ($t \in [t_{k-1}, t_{k-1} + T_{k-1}]$) and fix $\lambda \in [0, T_{k-1})$. Suppose the follower updates its trajectory only once during $[t_k, t_k + T_k]$ as described next:*

- *If $\lambda < T_{k-1} - \Delta$, the follower moves along the optimal trajectory (in the sense of (3)) joining $x_k(t_k + \lambda)$ and $x_{k-1}(t_k + \lambda)$ with optimal final time $\Gamma = \Gamma^*(x_k(t_k + \lambda), x_{k-1}(t_k + \lambda))$. During other times, the follower replicates the leader's trajectory, i.e.*

$$\begin{cases} x_k(t) = x_{k-1}(t - \Delta) & t \in [t_k, t_k + \lambda] \\ x_k(t) = x_{k-1}(t - \Gamma) & t \in [t_k + \lambda + \Gamma, t_k + T_k] \end{cases}$$

- *If $\lambda \geq T_{k-1} - \Delta$, the follower evolves along the optimal trajectory from $x_k(t_k + \lambda)$ to the constraint set S_Q (in the sense of (4)). Similarly, during other times*

$$x_k(t) = x_{k-1}(t - \Delta) \quad t \in [t_k, t_k + \lambda]$$

Then the cost along the follower's trajectory will be no greater than the leader's.

PROOF. First, choose $\lambda < T_{k-1} - \Delta$. Starting at time $t_k + \lambda$ and during $t \in [t_k + \lambda, t_k + \lambda + \Gamma]$, the follower moves on the locally optimal trajectory $x_k(t)$ (see Fig. 1). The cost along x_k is

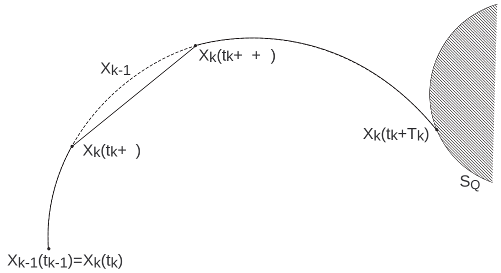

Fig. 1. Illustration of the trajectory obtained by a single update when $\lambda < T_{k-1} - \Delta$.

$$\begin{aligned} & C(x_k, t_k, T_k) = \\ & = C(x_k, t_k, \lambda) + C(x_k, t_k + \lambda + \Gamma, T_k - \lambda - \Gamma) \\ & \quad + \eta(x_k(t_k + \lambda), x_{k-1}(t_k + \lambda), t_k + \lambda) \\ & \leq C(x_{k-1}, t_{k-1}, \lambda) + C(x_{k-1}, t_{k-1} + \lambda, \Delta) \\ & \quad + C(x_{k-1}, t_{k-1} + \lambda + \Delta, T_{k-1} - \lambda - \Delta) \\ & = C(x_{k-1}, t_{k-1}, T_{k-1}) \end{aligned} \tag{6}$$

where $\Gamma = \Gamma^*(x_k(t_k + \lambda), x_{k-1}(t_k + \lambda))$.

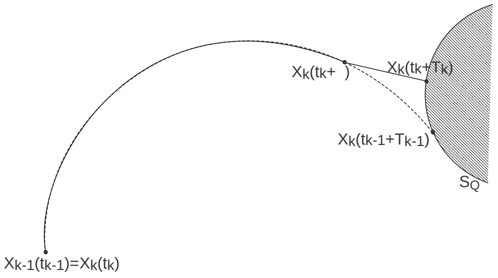

Fig. 2. Illustration of the trajectory obtained by a single update when $\lambda \geq T_{k-1} - \Delta$.

If $\lambda \geq T_{k-1} - \Delta$ (see Fig. 2), the cost along x_k is

$$\begin{aligned}
&C(x_k, t_k, T_k) = \\
&= C(x_k, t_k, \lambda) + \eta_Q(x_k(t_k + \lambda), t_k + \lambda) \\
&\leq C(x_{k-1}, t_{k-1}, \lambda) + C(x_{k-1}, t_{k-1} + \lambda, T_{k-1} - \lambda) \\
&= C(x_{k-1}, t_{k-1}, T_{k-1})
\end{aligned}$$

Therefore the cost along the follower's trajectory is no greater than the leader's. □

Now, the cost of the iterative trajectories can be shown to converge under mCLP:

Lemma 2 *(Convergence of Cost) If the agents (1) evolve under mCLP, the cost of the iterated trajectories converges.*

PROOF. Let C_{k-1} be the cost along the leader's trajectory $x_{k-1}(t)$ ($t \in [t_{k-1}, t_{k-1} + T_{k-1}]$). Define a trajectory sequence $x_k^i(t)$ ($t \in [t_k, t_k + T_k^i]$), $i = 0, 1, 2 \ldots,$ whose corresponding costs and final times are C_k^i and T_k^i, as follows: let $x_k^0(t) = x_{k-1}(t)$ (the trajectory of a "leader") and let x_k^i ($i > 0$) be the trajectory of an agent that pursues x_k^{i-1} by performing only a *single trajectory update*, as described in Lemma 1, with $\lambda = (i-1)\delta$, $\delta > 0$ (see Fig. 3).

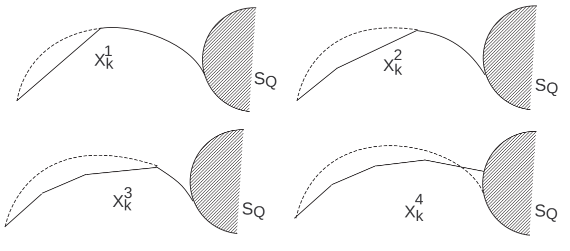

Fig. 3. Illustration of the trajectory sequence $x_k^i(t)$. Each trajectory is obtained by a single update upon its predecessor.

From Lemma 1, the cost of each follower's trajectory will be no greater than the leader's. Also, the sequence C_k^i is bounded below for fixed k. Thus, $C_k^i \leq C_k^{i-1}$ and $\lim_{k\to\infty} C_k^i = C_k^\infty$ exists for each k. Consequently,

$$C_k^\infty \leq C_k^0 = C_{k-1}$$

Now, take $\delta = T_{k-1}/i$, so that $\delta \to 0$ as $i \to \infty$. At the limit, the trajectory $x_k^\infty(t)$ is precisely what would be obtained by an agent that pursues its leader x_{k-1}, using mCLP. Hence, the follower's cost is $C_k = C_k^\infty \leq C_{k-1}$. Because the sequence $\{C_k\}$ is non-increasing and bounded below (there exists a minimum for (2)), it must converge to a limit. □

To proceed to the main theorem, we will require that the optimal cost of (2) changes "little" for small changes to the endpoints of a trajectory:

Condition 1 *Assume for a generic trajectory $x_1(t)$ there exists an $\varepsilon > 0$ such that for all $a, b_1, b_2 \in \mathbb{D}$ and all $\Omega > 0$, there exists a trajectory $x_2(t)$ such that the cost $C(x_1, 0, T)$ ($x_1(0) = a, x_1(T) = b_1$) from a to b_1 and cost $C(x_2, 0, T)$ ($x_2(0) = a, x_2(T) = b_2$) from a to b_2 satisfy*

$$\|b_1 - b_2\|_\infty < \varepsilon \Rightarrow \|C(x_1, 0, T) - C(x_2, 0, T)\|_\infty < \mathcal{L}\Omega$$

for some constant $\mathcal{L}$, independent of Ω.

Then the next lemma holds:

Lemma 3 *Let $x^*(t)$ be a trajectory of (1) such that: i) $x^*(t)$ ($t \in [0, t_1 + \Delta_1]$) is optimal (in the sense of (3)) from $x^*(0)$ to $x^*(t_1 + \Delta_1)$, and ii) $x^*(t)$ ($t \in [t_1, T^*]$) is optimal (in the sense of (4)) from $x^*(t_1)$ to the constraint set S_Q. Assume Condition 1 is satisfied and $0 < t_1 < t_1 + \Delta_1 < T^*$. Then the trajectory $x^*(t)$ ($t \in [0, T^*]$) is a local minimum of (4) from $x^*(0)$ to S_Q.*

PROOF. Choose $0 < \Delta \leq \Delta_1$. From the principle of optimality, $x^*(t)$ ($t \in [0, t_1 + \Delta]$) and $x^*(t)$ ($t \in [t_1, T^*]$) are each locally optimal with respect to their corresponding end points. Suppose $\|x^*(t_1 + \Delta) - s\|_\infty \geq \varepsilon_1$ for any $s \in S_Q$ and that $x^*(t)$ ($t \in [0, T^*]$) is not a local minimum. There must exist $\epsilon < \min(\varepsilon, \varepsilon_1/2)$ (where ε is defined in Condition 1) and another optimum $x(t) \in \mathbb{D} \times [0, T]$ satisfying $\|x(t) - x^*(t)\|_\infty < \epsilon$ and $C(x(t), 0, T) < C(x^*(t), 0, T^*)$.

Notice that $\|x(t_1 + \Delta) - s\|_\infty \geq \epsilon$ for any $s \in S_Q$. Construct two trajectories $y_1(t), y_2(t)$ ($t \in [t_1, t_1 + \Delta]$) that connect $x(t)$ and $x^*(t)$ (see Fig. 4) and satisfy Condition 1 (with x^* or x playing the role of x_1, and y_1 or y_2 standing in for x_2). In particular, let y_1, y_2 be such that $x^*(t_1) = y_2(t_1), x^*(t_1 + \Delta) = y_1(t_1 + \Delta), x(t_1) = y_1(t_1), x(t_1 + \Delta) = y_2(t_1 + \Delta)$. Now ,Condition 1 implies that

$$\begin{aligned}
&C(y_1(t), t_1, \Delta) < C(x(t), t_1, \Delta) + \mathcal{L}\Delta \\
&C(y_2(t), t_1, \Delta) < C(x^*(t), t_1, \Delta) + \mathcal{L}\Delta
\end{aligned} \tag{7}$$

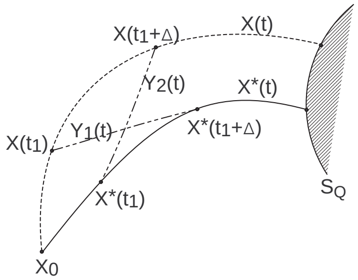

Fig. 4. Illustrating the proof of Lemma 3: "overlapping" optimal trajectories form a locally optimal trajectory.

Because $x^*(t)$ $(t \in [0, t_1 + \Delta])$ and $x^*(t)$ $(t \in [t_1, T^*])$ are each locally optimal, the following holds:

$$\begin{aligned} &C(x^*(t), 0, t_1) + C(x^*(t), t_1, \Delta) \\ &< C(x(t), 0, t_1) + C(y_1(t), t_1, \Delta), \text{ and} \end{aligned} \tag{8}$$

$$\begin{aligned} &C(x^*(t), t_1, \Delta) + C(x^*(t), t_1 + \Delta, T^* - t_1 - \Delta) \\ &< C(x(t), t_1 + \Delta, T - t_1 - \Delta) + C(y_2(t), t_1, \Delta) \end{aligned} \tag{9}$$

Combining (7) with (8,9) leads to

$$C(x^*(t), 0, T) < C(x(t), 0, T) + 2\mathcal{L}\Delta \tag{10}$$

The cost $C(x(t), 0, T)$ is apparently less than $C(x^*(t), 0, T)$; but if Δ is chosen so that

$$0 < \Delta < \frac{C(x^*(t), 0, T) - C(x(t), 0, T)}{2\mathcal{L}}$$

then (10) cannot hold. This is a contradiction, because Δ could be chosen arbitrarily small. It follows that $x^*(t)$ $(t \in [0, T^*])$ must be a local minimum. □

Assume that the locally optimal trajectory from the follower to the leader (or to S_Q) is unique at all times. This assumption is generally satisfied if pursuit is restricted to take place within a "small" region (setting Δ small), i.e. agents follow "close" to one another. Then, convergence of the trajectories' cost also implies convergence of the trajectories themselves:

Lemma 4 *If at all times during mCLP, the locally optimal trajectory from follower to leader (or to S_Q) is unique, then mCLP converges to a limiting trajectory $x_\infty(t)$.*

PROOF. Suppose that the trajectories' cost converges but that there exist more than one limiting trajectory. Let $x_1(t)$ $(t \in [0, T_1])$ and $x_2(t)$ $(t \in [0, T_2])$ be two such possibilities. Let $t_1 \in [0, T_1]$ be the earliest time that $x_1(t)$ differs from $x_2(t)$. From Lemma 2, x_1 and x_2 must have the same cost, otherwise convergence of the cost is contradicted. Suppose that a leader $x_{k-1}(t)$ travels along $x_1(t)$, while a follower $x_k(t)$ travels along $x_2(t)$. Choose $h > 0$ small, and that a series of sampled updates occur at $t_1 + ih$ $(i = 1, 2 \ldots, n = (T_1 - t_1 - \Delta)/h)$, as Fig. 5 indicates.

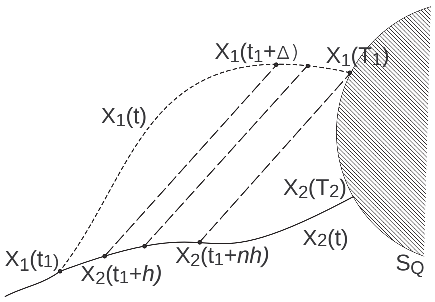

Fig. 5. Illustrating the proof of Lemma 4: pursuit between agents moving on two supposed "limiting" equal-cost trajectories, leads to the conclusion that the cost along the follower's trajectory is less than that along the leader's.

Consider the update occurring at t_1, the follower moves on $x_2(t), t \in [t_1, t_1+h)$ after this update. This fact means either that the trajectory passing $x_2(t), t \in [t_1, t_1 + h)$ and the optimal trajectory from $x_2(t_1+h)$ to $x_1(t_1+\Delta)$ (as indicated by the left dashed line in Fig. 5) has less cost than $x_1(t), t \in [t_1, t_1 + \Delta)$, or it has the same cost with $x_1(t), t \in [t_1, t_1 + \Delta)$. The latter is contradict to the assumption that there only exists a unique locally optimal trajectory from follower to leader at any time. Therefore the locally optimal trajectory that the follower actually calculated at t_1 has the cost of $C(x_2, t_1, h) + \eta(x_2(t_1 + h), x_1(t_1 + \Delta), t_1 + h)$, and

$$\begin{aligned} &C(x_2, t_1, h) + \eta(x_2(t_1 + h), x_1(t_1 + \Delta), t_1 + h) \\ &< C(x_1, t_1, \Delta) \end{aligned} \tag{11}$$

Similarly, investigate the update occurring at $t_1 = ih(i = 2, 3, \ldots, n)$, and we obtained

$$\begin{aligned} &\quad C(x_2, t_1 + h, h) \\ &\quad +\eta(x_2(t_1 + 2h), x_1(t_1 + \Delta + h), t_1 + 2h) \\ &< \eta(x_2(t_1 + h), x_1(t_1 + \Delta), t_1 + h) \\ &\quad +C(x_1, t_1 + \Delta, h) \end{aligned} \tag{12}$$

$\ldots\ldots$

$$\begin{aligned} &\quad C(x_2, t_1 + (n-1)h) \\ &\quad +\eta(x_2(t_1 + nh), x_1(T_1), t_1 + nh) \\ &< \eta(x_2(t_1 + (n-1)h), x_1(T_1 - h), t_1 + (n-1)h) \\ &\quad +C(x_1, T_1 - h, h) \end{aligned} \tag{13}$$

And at the last update step, the follower will choose the locally optimal trajectory from itself to S_Q:

$$\begin{aligned} &C(x_2, t_1 + nh, T_2 - t_1 - nh) \\ &< \eta(x_2(t_1 + nh), x_1(T_1), t_1 + nh) \end{aligned} \quad (14)$$

Notice the inequality does not depends on the size of h. No matter how small h is, it can always be concluded from (11)$\sim$(14) that

$$C(x_2, t_1, T_2 - t_1) < C(x_1, t_1, T_1 - t_1) \quad (15)$$

If we let $h \to 0$, the above sampled process will approach the continuous local pursuit. However, the result of (15) does not change with the decrease of h, therefore the cost along $x_2(t)$ must be less than that for $x_1(t)$ under mCLP, which contradicts the convergence of the cost under mCLP. □

Lemma 5 *Along the limiting trajectory produced under mCLP, the planned trajectories $\hat{x}_k(t)$ and realized trajectories $x_k(t)$ overlap, i.e. $\hat{x}_k(t) = x_k(t)$. Furthermore, if the locally optimal trajectories obtained at every updating time are smooth, then the limiting trajectory is also smooth.*

PROOF. Suppose that a leader, x_{k-1} evolves along the limiting trajectory $x_\infty(t)$. Lemma 4 then implies that $x_{k-1}(t) = x_k(t + \Delta)$ for $\forall t \in [t_k, t_k + T_k]$.

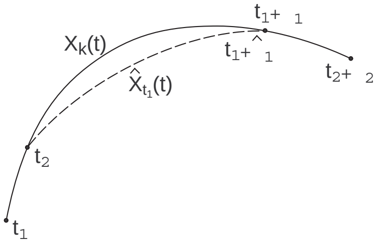

Fig. 6. Differences between the planned and realized trajectories contradict the convergence of trajectories under mCLP.

Suppose that with the leader at $x_{k-1}(t_1 + \Gamma(t_1))$, where $\Gamma(t_1)$ is the best final time for update at t_1, and follower at $x_k(t_1)$, the planned trajectory $\hat{x}_{t_1}(t)$ ($t \in [t_1, t_1 + \hat{\Gamma}(t_1))$ obtained at t_1 differs from $x_k(t)$ ($t \in [t_1, t_1 + \Gamma(t_1))$ starting at some time $t_2 \geq t_1$. Furthermore, let $\hat{x}_{t_1}(t_1 + \hat{\Gamma}(t_1)) = x(t_1 + \Gamma(t_1))$. Because the planned trajectory $\hat{x}_{t_1}(t)$ is unique (by assumption) and optimal,

$$C(\hat{x}_{t_1}, t_2, \hat{\Gamma}(t_1) - (t_2 - t_1)) < C(x_k, t_2, \Gamma(t_1) - (t_2 - t_1))$$

Construct the trajectory

$$\bar{x}(t) = \begin{cases} \hat{x}_{t_1}(t) & t \in [t_2, t_1 + \hat{\Gamma}(t_1)) \\ x_k(t - \hat{\Gamma}(t_1) + \Gamma(t_1)) & t \in [t_1 + \hat{\Gamma}(t_1), t_2 + \Gamma(t_2)] \end{cases}$$

Clearly, $\bar{x}$ has lower cost than $x_k(t)$ ($t \in [t_2, t_2 + \Gamma(t_2)]$) (See Fig. 6). Thus, under mCLP, the follower would have taken $\bar{x}$ (or another trajectory with even lower cost) over $x_k(t)$ ($t \in [t_2, t_2 + \Gamma(t_2)]$). This contradicts the convergence to a limiting trajectory. The same argument can be applied at any other updating time, so that it can be concluded that $\hat{x}(t) = x_k(t)$ ($t \in [0, T_k]$).

Recall that $x_k(t)$ is smooth for $t \in [t_1, t_1 + \Gamma(t_1)]$, because the locally optimal trajectories linking follower and leader are smooth by assumption. Similarly, $x_k(t)$ is smooth for $t \in [t_2, t_2 + \Gamma(t_2)]$ for any $t_1 < t_2 < t_1 + \Gamma(t_1)$. Therefore, $x_k(t)(t \in [t_1, t_2 + \Gamma(t_2)])$ is smooth. Repeated applications of this argument lead to the conclusion that the entire trajectory $x_k(t)$ ($t \in [0, T_k]$) is smooth. □

The next theorem is an immediate consequence of Lemmas 1 $\sim$ 5:

Theorem 1 *Suppose that the group of (1) evolves under mCLP and that at all times t, the locally optimal trajectories from follower to leader are unique. Then, the limiting trajectory is unique and locally optimal. It is also smooth, if the locally optimal trajectories calculated at every updating time are smooth.*

PROOF. From Lemma 4, the limiting trajectory is unique. It follows that $x_{k-1}(t - \Delta) = x_k(t)$ if $x_{k-1}(t) = x_\infty(t - t_{k-1})$. Choose δ_1, δ_2 such that $0 < \delta_1 < \delta_2 < \Gamma$ for all optimal final times Γ of the planned trajectories $\hat{x}_k$ generated during mCLP. The limiting trajectory x_∞ is piecewise smooth and locally optimal for $t \in [t_k + i\delta_1, t_k + i\delta_1 + \delta_2], i = 0, 1, 2\ldots$ because it coincides with the planned trajectories $\hat{x}_k(t)$. From Lemma 3 – in this case S_Q is a single point – it can be concluded that $x_k(t)$ ($t \in [t_k, t_k + \delta_1 + \delta_2]$) is optimal because it is the composition of two overlapping locally optimal trajectories, $x_k(t)$ ($t \in [t_k, t_k + \delta_2]$) and $x_k(t)$ ($t \in [t_k + \delta_1, t_k + \delta_1 + \delta_2]$). ¿From successive applications of this argument ($i = 2, 3, \ldots$), we conclude that $x_\infty(t)$ is locally optimal. Smoothness of x_∞ is proved via a similar "piece by piece" argument. □

3.1 Remarks

Local pursuit is a cooperative, decentralized algorithm for learning optimal controls/trajectories, starting from a feasible solution. Each agent is only required to calculate optimal trajectories from its own state to that of its nearby leader. Because agents are separated by Δ time units as they leave x_0, each agent relies on local information only in order to follow its predecessor, and requires no knowledge of the global geometry. Therefore there is no need for agents to exchange or "fuse" local maps that they obtain individually. Agents do not need to communicate their choice of coordinate systems as they evolve, nor do they need to know the coordinates of x_f. While it is possible that a group of agents could disperse and construct a global map from local information, such an approach might require significantly more computation and communication than local pursuit. The latter solves the optimal control problem in many "short pieces", which makes it no need to compute the optimum over the whole environment. Thus local pursuit is appropriate for systems with short-range sensors (for example, in the case of a swarm of robots exploring unknown terrain), and optimal control problems which are easier to solve over "short" distances.

The local pursuit algorithms assumed a countable infinity of agents; of course, such a collection cannot be realized. It is however possible to achieve the same results with a finite number of agents that apply local pursuit to reach the final constraint set S_Q from x_0, then return to x_0 along the obtained path. The required modifications are straightforward but will not be discussed here as they are beyond the scope of this report. An experiment that uses this technique is detailed in [8]. Finally, local pursuit is not guaranteed to converge to the global optimum. The choice of agent separation Δ can affect whether the limiting trajectory is a local or a global optimum. Some interesting cases involving spaces with holes or obstacles are discussed in [8,14].

4 Simulations and Experiments

In this section, we describe a series of simulations and an experiment desinged to illustrate the performance of local pursuit.

4.1 A trail optimization problem with free final states

Consider the problem of finding shortest paths in an environment consisting of a plane with two right cones, whose top view was shown in Fig. 7. The radii and heights of the cone were 800 and 1000 units of length, respectively. Each object (the plane and each cone) was parametrized with its own set of coordinate functions. The agents were governed by $\dot{x}_k = u_k, \|u_k\| = 1$ and were required to travel from $x_0 = (3500, 0, 0)$ to the second cone.

Fig. 7 shows the iterated trajectories generated by a collection of systems implementing the mCLP policy with $T_0 = 3499,\ \Delta = 0.2T_0$. For the computation of the optimal trajectory, each agent had to solve its own optimal control problem which was simpler than the "global" problem, partly due to the fact that the optimal trajectory crosses multiple coordinate patches as it crosses from the plane to the cones and vise versa. When leader and follower were both on the plane or on the same cone, the computation of optimal trajectories was straightforward. In other cases, agents had to compute optimal trajectories that crossed between at most two coordinate patches (plane-to-cone or cone-to-plane). On the other hand, computing the optimal trajectory at once would require searching over a four-parameter family of curves (there are a total of four "crossings" between coordinate sets). A thorough accounting of the computational requirements and numerical performance of local pursuit will be forthcoming.

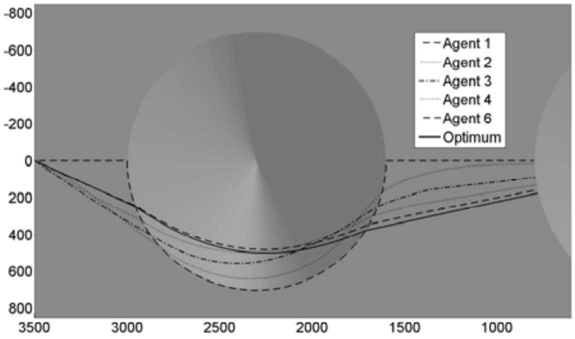

Fig. 7. Continuous local pursuit in a complex environment. The initial trajectory (along the borders of the cones) is easily described but far away from optimal. The locally optimal trajectories were easier to compute than the global optimum because of the limited pursuit distance ($\Delta = 0.2T_0$). The iterated trajectories converged to the optimum.

4.2 Minimum-time control with limited acceleration and speed

Next, consider the minimum-time control of the second-order system

$$\ddot{x} = u; \quad s.t. \ \ |u| \le 30, \ |\dot{x}| \le 8$$

We want to minimize $J(x, \dot{x}, 0) = T$, with the boundary conditions $\dot{x}(0) = \dot{x}(T) = 0$, $x(0) = 0$ and $x(T)$ fixed (in this simulation, $x(T)$ is determined by the input to the first agent). Here the constraint set S_Q is a single point in the state space. The optimal control policy is similar to the well-known 'bang-bang" control: the control u switches at most once between 30 and -30, and $u = 0$ when the maximum or minimum speed $\dot{x}$ has been reached. The initial, suboptimal input (Agent 1 in Fig. 8), alternated between the maximum and minimum available acceleration. When using mCLP with $\Delta = 1.3$sec, the third agent's trajectory was optimal,

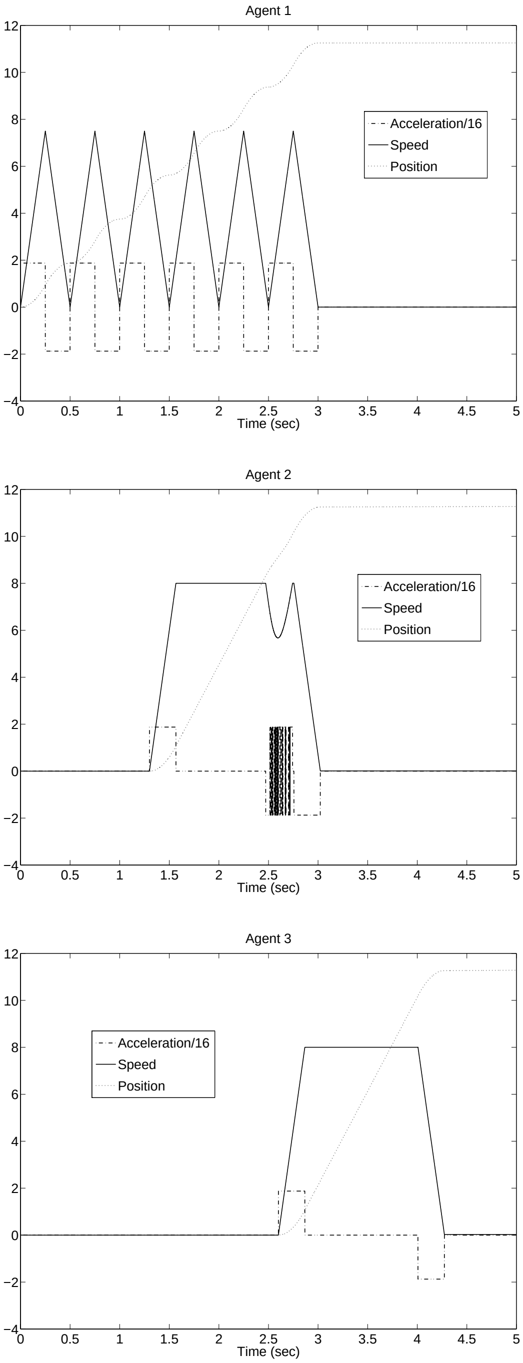

Fig. 8. Iterative trajectories for minimum control with limited acceleration and speed. The simulated control loop ran at a frequency of $2000Hz$ so that the control policy could be regarded as approximately mCLP. The pursuit interval was $\Delta = 1.3$. Units for acceleration, velocity and position are $Rad/s^2, Rad/s, Rad$, respectively.

see Fig. 8 for illustration. Notice that after $t > 2.7sec$ the second agent intercepted the first and subsequently moved along the same trajectory x_1. It is also interesting to note that in this case, optimality was achieved after a finite number of iterations.

4.3 Experiment on minimum-time control with acceleration and speed constraints

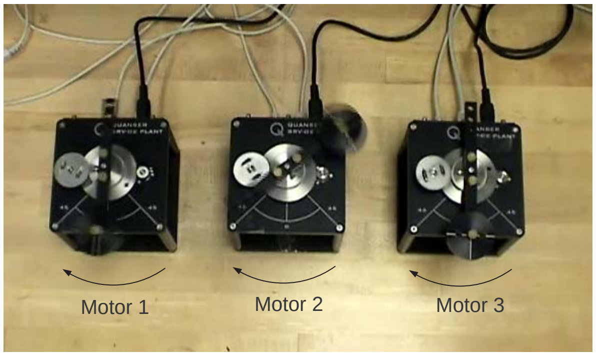

Fig. 9. Applying local pursuit with a trio of motors to obtain minimum-time control with limited acceleration and speed.

We implemented the example of Sec. 2.4 using a collection of three motors, shown in Fig. 9. Each motor was equipped with position and speed sensors, which were sampled by a PC-based controller at a rate of $2000Hz$. The goal was to rotate the motors to a fixed final position in minimum time. Motor acceleration and speed were limited to $30rad/sec^2$ and $8rad/sec$, respectively.

The input to the first motor was a rectangular pulse with amplitude equal to the maximum acceleration (same as in the simulation of Sec. 2.4). Each of the remaining two motors tried to "catch up" with its predecessor by reaching the predecessor's state minimum time. The trajectories of all three motors with $\Delta = 1.3$sec are shown in Fig. 10. We see that the third motor evolved under essentially optimal control, and the second motor "intercepted" the first after $t \approx 2.3sec$.

Because of unmodeled friction, the final position $\theta(T)$ was less than the nominal value (see $x(T)$ in the last simulation). Friction also caused the motors to decelerate when a zero input was applied (once the motors reached maximum speed). In turn, that deceleration caused the mCLP policy to try and catch up by introducing a positive control input, resulting in chatter observed in the velocity and acceleration curves of motors 2 and 3 in Fig. 10.

5 Conclusions and ongoing work

This report explored a biologically-inspired cooperative strategy (termed "Local Pursuit") for solving a class of optimal control problems with free final time

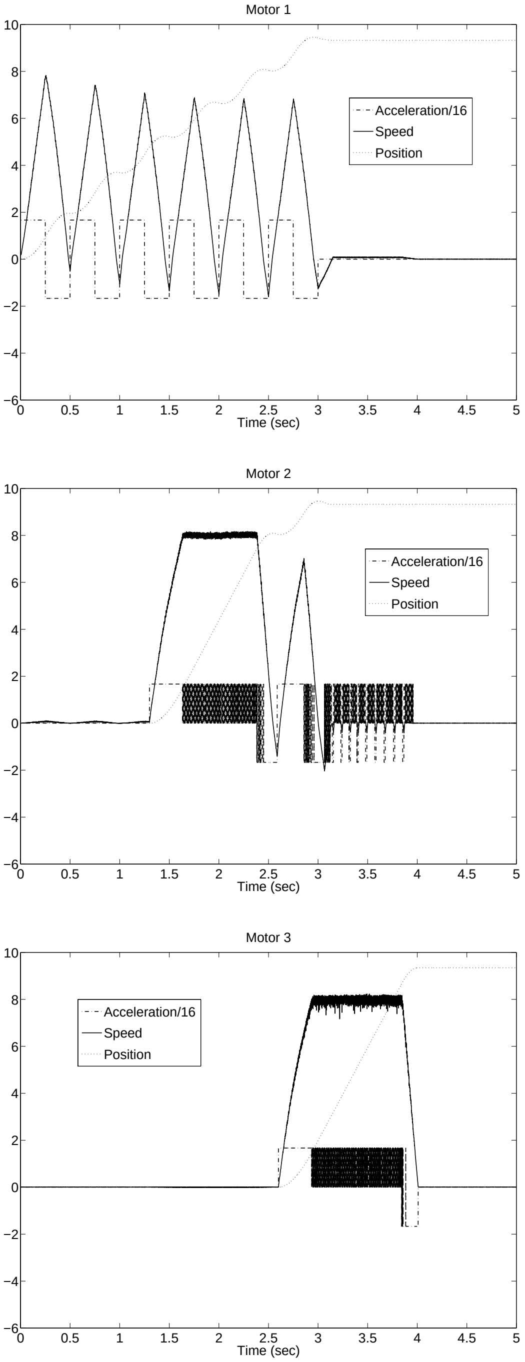

Fig. 10. Iterative trajectories of motors when applying local pursuit to attain minimum-time control with limited acceleration and speed. The pursuit interval $\Delta = 1.3$. The third motor evolved under essentially optimal control.

and partially-constrained final state. The proposed algorithms generalizes previous models that mimic the foraging behavior of ant colonies and allows a collective to discover optimal controls, starting from an initial suboptimal solution. Members of the collective are only required to obtain local information on their environment and to calculate optimal trajectories to their nearby neighbors. The local pursuit algorithm relies on cooperation to perform a task which would be difficult or impossible for a single system to perform, namely solving an optimal control problem with limited information (in terms of coordinate systems that describe the environment or the coordinates of the final state) and short-range sensing.

Although this work was inspired by a desire to explore the limits of a simple-to-formulate, bio-inspired control policy, mCLP and especially its “sampled” counterpart could be interesting as numerical methods for computing optimal controls. Work in that direction is ongoing.

References

[1] R.A. Brooks and A.M Flynn. Fast, cheap and out of control: a robot invasion of the solar system. *Journal of the British Interplanetary Society*, 42:478–485, 1989.

[2] A.M. Bruckstein. Why the ant trails look so straight and nice. *The Mathematical Intelligencer*, 15(2):59–62, 1993.

[3] A.M. Bruckstein, C.L. Mallows, and I. A. Wagner. Probabilistic pursuits on the grid. *The American Mathematical Monthly*, 104(4):323–343, April 1997.

[4] S. Camazine, J.-L. Deneubourg, N. R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. *Self-Organization in Biological Systems*. Princeton University Press, Princeton, New Jersey 08540, 2001.

[5] M. Dorigo, V. Maniezzo, and A. Colorni. Ant systems: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 26(1):29–41, 1996.

[6] R. Fierro, P. Song, A. Das, and V. Kumar. Cooperative control of robot formation. In R. Murphey and P.M. Pardalos, editors, *Cooperative control and optimization*, pages 73–93. Kluwer Academic Publishers, 2002.

[7] D. Hristu-Varsakelis. Robot formations: Learning minimum-length paths on uneven terrain. In *Proceedings of the 8th IEEE Mediterranean Conference on control and Automation*, 2000.

[8] D. Hristu-Varsakelis and C. Shao. Biologically-inspired optimal control: Learning from social insects. *International Journal of Control*, 77(18):1545–66, Dec. 2004.

[9] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Transactions on Automatic Control*, 48(6), June 2003.

[10] R. Kurazume and S. Hirose. Study on cooperative positioning system: optimum moving strategies for cps-iii. In *Proceeding of the 1998 IEEE International Conference in Robotics and Automation*, volume 4, pages 2896–2903, Leuven, Belgium, May 1998.

[11] P. Ögren, E. Fiorelli, and N.E. Leonard. Formation with a mission: Stable coordination of vehicle group maneuvers. In *Proceedings Fifteenth International Symposium on Mathematical Theory of Networks and Systems*, Notre Dame, IL, August 2002.

[12] K. Passino, M. Polycarpou, D. Jacques, M. Pachter, Y. Liu, Y. Yang, M. Flint, and M. Baum. Cooperative control for autonomous air vehicles. In R. Murphey and P.M. Pardalos, editors, *Cooperative control and optimization*, pages 233–272. Kluwer Academic Publishers, 2002.

[13] S.I. Roumeliotis and G.A. Bekey. Distributed multi-robot localization. *IEEE Transactions on Robotics and Automation*, 18(5):781–795, 2002.

[14] C. Shao and D. Hristu-Varsakelis. Biologically inspired algorithms for optimal control. Technical Report TR2004-29, Institute for Systems Research, University of Maryland, College Park, MD 20742, 2004.

[15] T. Vicsek, A. Czirok, E.B. Jacob, I. Cohen, and O. Schochet. Novel type of phase transitions in a system of self-driven particles. *Physical Review Letters*, 75:1226–1229, 1995.

[16] I.A. Wagner, M. Lindenbaum, and A.M. Bruckstein. Distributed covering by ant-robots using evaporating traces. *IEEE Transactions on Robotics and Automation*, 15(5):918–933, 1999.

[17] H. Yamaguchi and J.W. Burdick. Asymptotic stabilization of multiple nonholonomic mobile robots forming group formations. In *Proceedings of IEEE International Conference on Robotics and Automation*, volume 4, pages 3573–3580, 1998.